

Grau en Matemàtiques

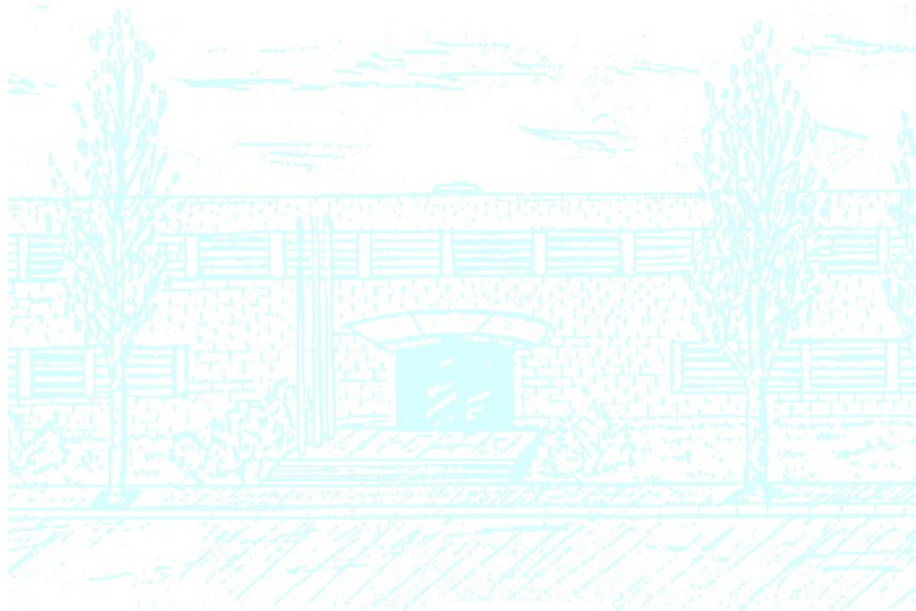
Títol: Continuation Methods: Application to a Biophysical Problem

Autor: Ivan Caro Mañas

Director: Blas Echebarria

Departament: Física

Convocatòria: Extraordinaria



Universitat Politècnica de Catalunya
Facultat de Matemàtiques i Estadística

Degree in Mathematics

CONTINUATION METHODS: APPLICATION TO A BIOPHYSICAL PROBLEM

Ivan Caro Mañas

Supervised by Blas Echebarria Dominguez

3 de setembre, 2019

Moltes gràcies al Blas Echebarria per les hores que ha invertit amb reunions amb mi, ajudant-me i guiant-me en l'elaboració d'aquest treball.

Abstract

The goal of this project is to study continuation methods and to learn how to implement them numerically. We will review the main points of the procedure and we will try to replicate these results numerically. Starting with some concepts learned during the degree like definitions or theorems, for example the *Implicit function Theorem*, we will study basic concepts about curves to use them in solution techniques applying deformation, as well as conditions they have to satisfy to prevent the curve from running to infinity. Assuming that, we will undergo a study about predictor-corrector(PC) methods, which are the main objective of this project, understanding how they work and programming them in a software, in our case, MATLAB. Finally we will take advantage of the codes that we have programmed, to apply PC methods in a more particular case and draw some conclusions of it. The case that we have considered is a model of a paced cardiac cell with cytosolic and luminal spaces, each separated into different compartments, i.e dyadic, subsarcolemmal and cytosolic. We calculate the type of solution that we find as we change the total amount of calcium in the system and we discuss some conclusions of these solutions.

1. Introducció

Els mètodes de continuació són tècniques per a produir seqüències de solucions molt properes d'un conjunt d'una o més equacions algebraïques. Podem imaginar-ho com un conjunt de punts geomètrics que tracen una corba.

En el cas simple on només tenim una equació, podem escriure-la de forma implícita i traçar la corba de solucions sense la necessitat d'aplicar cap tècnica. Ara bé, sempre trobarem equacions en les que ens plantejarem tot un seguit de preguntes. Existeixen solucions de l'equació? Si existeixen, es pot traçar una corba de solucions molt properes? És una corba tancada o en algun moment va cap a l'infinit? Els mètodes de continuació són usats en aquestes situacions complicades donant-nos una via per a produir ordenadament seqüències de solucions un cop arribem a aquest punt dubitatiu.

Des del passat, dues tècniques van esdevenir extremadament importants per a la contribució de trobar solucions de sistemes no lineals. Aquestes dues tècniques o mètodes de continuació han anat rebent varis noms amb el pas del temps, un d'ells ha estat conegut com *predictor-corrector*, mentre que l'altre es referit com a *piecewise linear method*.

Els mètodes de continuació o homotopies han sigut molt útils en les matemàtiques modernes, desde Poincaré (1881-1886) fins a Lahaye (1934) qui va començar a utilitzar les deformacions per a resoldre sistemes d'equacions no lineals. Actualment, els mètodes classics per a resoldre sistemes podrien considerarse com a precursors del mètode *predictor-corrector method*, el qual tractarem exhaustivament i en el que basarem aquest estudi.

Tot i que ambdós mètodes són diferenciats, també tenen coses en comú i estan basats en alguns principis generals similars. Així doncs, l'objectiu d'aquest treball serà investigar com funcionen els mètodes de continuació, estudiant i replicant els processos, introduint els conceptes claus per a comprendre els articles i reproduint-ne els càlculs tant analítics com computacionals. Estudiarem les condicions que hauran de complir les corbes formades a partir de les seqüències de solucions dels sistemes d'equacions.

Per a realitzar aquest estudi ens basarem en varis articles sobre mètodes de continuació i intentarem aplicar els processos de manera numèrica. Així doncs, utilitzarem MATLAB per intentar replicar els processos amb la finalitat de tindre el procés automatitzat.

Finalment, aplicarem tots els conceptes apresos sobre els mètodes de continuació en un cas biofísic per a poder-ne veure més clarament la utilitat i poder extreure conclusions de la seva aplicació.

2. Descripció del model

2.1 Conceptes previs

Primerament, per a situarnos, un mètode d'homotopia consisteix en el següent. Suposem que desitgem trobar la solució d'un sistema de N equacions no lineals amb N variables, sigui:

$$F(x) = 0 \quad \text{on} \quad F(x) : R^N \rightarrow R^N. \quad (1)$$

Suposem que $F \in C^1$ per a facilitar les coses. Considerem ara, que a priori tenim poc coneixement sobre els zeros de la funció F . Si tenim una bona aproximació x_0 de un zero de F , un bon mètode per a trobar zeros de la funció F és mitjançant l'algoritme iteratiu de Newton definit per la fórmula:

$$x_{i+1} := x_i - A_i^{-1} F(x_i) \quad \text{on} \quad i = 0, 1, \dots \quad (2)$$

on A_i és una aproximació numèrica de la Jacobiana de F en el punt $x_i : F'(x_i)$. Ara bé, això només ho podem executar si coneixem a priori un punt prou proper a la solució, sinó, el mètode de Newton pot fallar. Com a solució d'aquest problema, es planteja la homotopia de deformació $H : R^N \times R \rightarrow R^N$ que compleix:

$$H(x, 1) = G(x), \quad H(x, 0) = F(x), \quad (3)$$

on $G \in C^1$ és una funció $R^N \rightarrow R^N$ de la qual en coneixem els zeros i H és també continua i derivable. Un cop definits aquests termes, definim la homotopia convexa:

$$H(x, \lambda) := \lambda G(x) + (1 - \lambda) F(x), \quad (4)$$

A partir d'aquesta homotopia, intentarem resseguir una corba implícita definida $c(s) \in H^{-1}(0)$ des d'un punt inicial $(x_1, 1)$ fins a la solució $(\bar{x}, 0)$. Si arribem a aquesta solució, aquesta \bar{x} serà un zero de F .

Després de comprovar aquest fet, ens qüestionarem:

1. Quan podem assegurar que la corba $c(s) \in H^{-1}(0)$ existeix i és continua i derivable?
2. En el cas que existeixi la corba, quan podem assegurar que tallarà l'altre extrem de l'homotopia en el nivell $\lambda = 0$?
3. Com podem fer aquest procediment numèricament?

La primera pregunta és resposta mitjançant el teorema de la funció implícita. Si recordem aquest teorema:

Teorema 2.1 (Teorema de la Funció Implícita)

Sigui $H : R^N \times R \rightarrow R^N$ satisfent:

- $H(x_0, \lambda_0) = 0, \quad x_0 \in R^N, \lambda_0 \in R$

- $H_x(x_0, \lambda_0) = 0$ no singular
- H i H_x funcions contínues en un entorn proper a x_0

Aleshores existeix una única família de solucions $x(\lambda)$ tal que:

- $H(x(\lambda), \lambda) = 0$, per a tot λ proper a λ_0
- $x(\lambda_0) = u_0$

Així doncs, nomenant $(x_1, 1)$ com a un zero regular de H , és a dir, en el qual la Jacobiana de H té rang màxim, aleshores existeix una corba $c(s) \in H^{-1}(0)$ amb condició inicial $c(0) = (x_1, 1)$ i derivada $c'(0) \neq 0$ en algun interval al voltant del zero.

La segona resposta va lligada amb els teoremes d'existència en el anàlisi d'equacions no lineals. Generalment, és suficient amb aplicar alguna condició de contorn de tal manera que s'eviti que la corba arribi a infinit abans que intersecti amb el nivell $\lambda = 0$ o que retorni al nivell $\lambda = 1$. Més endavant veurem com aplicar aquestes condicions.

Per últim, la tercera pregunta és la que investigarem més a fons, és a dir, com podem trobar els zeros de la funció numèricament sense tindre una aproximació inicial.

2.2 Mètode de incrustació

El següent algoritme va ser inicialment estudiat per Ficken(1951) en el seu article sobre els mètodes de continuació [4]. La idea general ve donada per traçar una corba agafant com a paràmetre de continuació cert λ . És a dir, suposem que tenim una solució (x_0, λ_0) de l'equació (4). Aleshores, volem trobar una solució (x_1, λ_1) on $\lambda_1 = \lambda_0 + \Delta\lambda$.

Començant a $\lambda = 1$, en un punt x que sabem que és solució: $H(x, 1)$, anirem variant el λ amb un pas $\lambda_i = \lambda_{i-1} + \Delta\lambda$ fins arribar $\lambda = 0$. Així doncs, en pseudo-codi vindria a ser:

Mètode d'incrustació

$$\begin{aligned}
 &x_1 \in \mathbf{R}^N \quad \text{on} \quad H(x_1, 1) = 0 \\
 &\quad \text{Agafem} \quad n > 0 \\
 &\text{Definim} \quad \lambda := (n-1)/n; \Delta\lambda := 1/n; \\
 &\quad i = 1, \dots, n \\
 &\text{Resolem} \quad H(y, \lambda) = 0; \\
 &\quad x := y; \lambda := \lambda - \Delta\lambda
 \end{aligned}
 \tag{5}$$

Cal tindre en compte, que per a resoldre $H(y, \lambda) = 0$ farem servir un mètode iteratiu com per exemple Newton, ja que al tindre un punt inicial proper a la solució, convergirà ràpidament. Així doncs, la idea d'aquest algoritme és clara: a partir de petits increments de λ , ens anirem apropant a la solució esperada $H(x, 0)$ i a causa d'agafar aquests petits increments, els punts

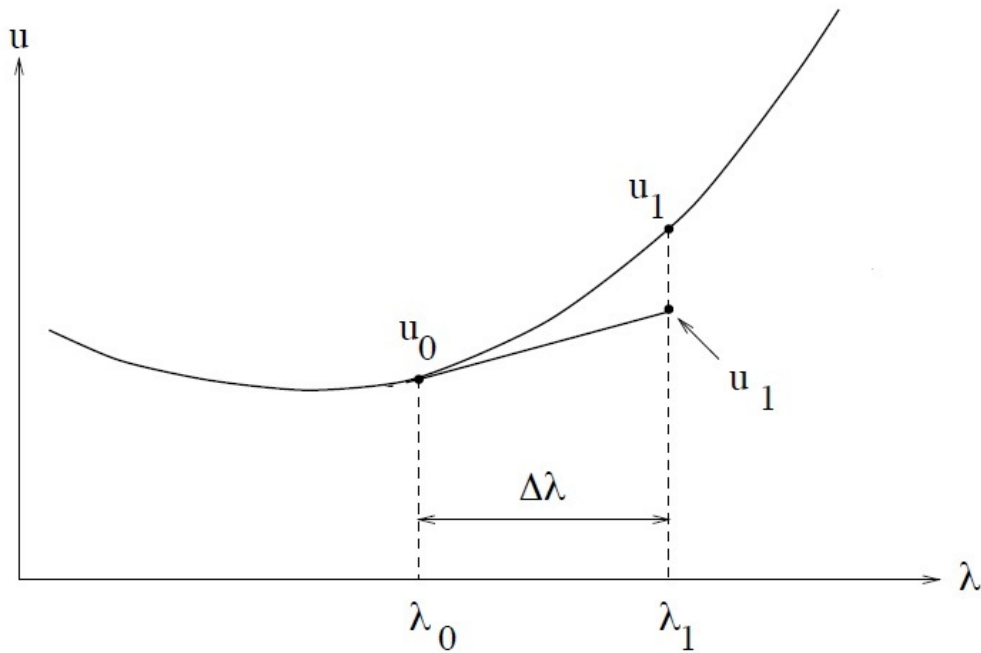


Figura 1: Foto extreta de l'article [1]. Podem veure que partint del punt $u_0 = (x_0, \lambda_0)$, en la següent iteració del mètode d'incrustació trobem el punt $u_1 = (x_1, \lambda_1)$.

seran prou propers i per tant, el procés convergirà. Podem veure gràficament el procés del mètode d'incrustació en la figura 1.

El problema del mètode és que fallarà en corbes en les qual hi hagi un retrocés enrere com podem comprovar en la següent imatge, ja que aleshores saltarà a una corba que és diferent de la que estem resseguint; com es pot comprovar en la figura 2.

Un remei per a aquest problema és resseguir la corba a partir del paràmetre arc. Així doncs, la corba c , serà parametritzada per el paràmetre arc s i serà solució d'un problema de valor inicial:

$$H(c(s)) = 0 \quad (6)$$

on respecte s s'ha de complir:

$$H'(c)\dot{c} = 0, \quad \|\dot{c}(s)\| = 1, \quad c(0) = (x_1, 1) \quad (7)$$

Així doncs, veiem quines condicions s'han de satisfer per a poder definir aquesta corba $c(s)$.

2.3 Corbes definides implícitament

En la introducció anterior ens hem situat en el context per a trobar solucions del sistema $H(x, \lambda) = 0$. Hem vist que per a resoldre-ho definirem una corba $c(s)$ de punts solucions. En aquesta secció descriurem condicions sobre les corbes.

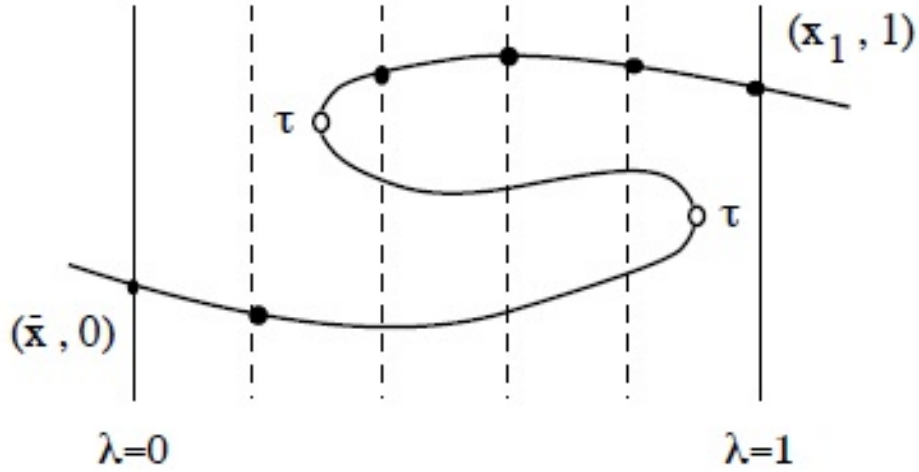


Figura 2: Figura extreta de [2]. Els punts τ són aquells en els que el mètode d'incrustació fallaria, degut al canvi de sentit de la corba.

Sigui $H : R^{N+1} \rightarrow R^N$ una funció continua, direm continua per eliminar hipòtesis complicades, suposem que existeix un punt $u_0 = (x_0, \lambda_0)$ tal que

- $H(u_0) = 0$
- La matriu Jacobiana $H'(u_0)$ té rang màxim, és a dir, $\text{rang}(H'(u_0)) = N$.

Lema 2.2

Sota les condicions anterior, aleshores existeix una corba $\alpha \in I \rightarrow c(\alpha) \in R^{N+1}$ per a algun interval I que conté el zero.

1. $c(0) = u_0$;
2. $H(c(\alpha)) = 0$
3. $\text{rang}(H'(c(\alpha))) = N$
4. $c'(\alpha) \neq 0$

De la segona equació del Lema (2.2) podem concloure que la seva tangent $c'(\alpha)$ compleix l'equació:

$$H'(c(\alpha))c'(\alpha) = 0 \quad (8)$$

I per tant, això ens indica que $c'(\alpha)$ és ortogonal a totes les files de la matriu $H'(c(\alpha))$. Com a conseqüència d'això, és convenient parametritzar la corba respecte el paràmetre arc s de la manera següent:

$$\left[\sum_{i=1}^{N+1} \left(\frac{dc_j(\alpha)}{d\alpha} d\alpha \right)^2 \right]^{1/2} = ds$$

on c_j representa la j -èssima posició de c .

Així doncs, substituint el paràmetre α pel nou paràmetre arc, tenim que:

$$\|\dot{c}(s)\| = 1, s \in I$$

per a algun interval I , on la derivada de c queda definida per:

$$\dot{c} = \frac{dc}{ds},$$

El nucli de la Jacobiana $H'(c(s))$ té exactament dos vectors unitaris que corresponen a les dues possibles direccions d'atravessar la corba. En general, un desitja recórrer la corba sempre en una única direcció, així doncs, introduïm el lema següent:

Lema 2.3

Sigui $c(s)$ la solució amb orientació positiva de la corba parametritzada pel paràmetre arc respecte s , tal que satisfà $c(0) = u_0$ i $H(c(s)) = 0$ per a s pertanyent a un interval I que conté el zero. Aleshores, per a tot $s \in I$, la tangent $\dot{c}(s)$ satisfà les 3 condicions següent:

1. $H'(c(s))\dot{c}(s) = 0$
2. $\|\dot{c}(s)\| = 1$
3. $\det \begin{pmatrix} H'(c(s)) \\ \dot{c}(s)^* \end{pmatrix} > 0$ on $\dot{c}(s)^*$ representa la transposada de la tangent de c .

Definits els conceptes anteriors, el que volem trobar és una única corba $c(s)$ que sigui solució del problema inicial següent:

Definició 2.4. Problema del Valor Inicial

1. $\dot{u} = t(H'(u))$
2. $u(0) = u_0$

on t representa la tangent. Seguidament, coneixerem un seguit de definicions i teoremes a partir dels quals podrem assegurar l'existència d'aquesta corba.

Definició 2.5. Sigui $f : R^M \rightarrow R^N$ una funció continua.

1. Diem que $x \in R^M$ és un punt regular de f si la Jacobiana $f'(x)$ té rang màxim ($\min(M, N)$).
2. Diem que $y \in R^N$ és un valor regular de f si x és un punt regular de f per a tot $x \in f^{-1}(y)$.
3. Diem que un punt o valor és singular, si no és regular.

El següent lema mostra que tots els punts de la corba solució són zeros regulars de H , i per tant, no tindrem inconvenients al fer els càlculs quan implementem el mètode.

Lema 2.6

Si $a > -\infty$ aleshores la corba $c(s)$ convergeix cap a un punt \bar{u}_1 quan $s \rightarrow a$, $s > a$, el qual és un zero singular de H . De la mateixa manera, si $b < \infty$ sigui $s < b$, la corba $c(s)$ convergeix cap a un punt \bar{u}_2 quan $s \rightarrow b$, el qual és un zero singular de H .

Finalment, podem donar peu al punt més rellevant de tota aquesta petita introducció a les corbes.

Teorema 2.7

Sigui zero un valor regular de $H = \lambda F(x) + (1 - \lambda)G(x)$. Aleshores la corba c és definida a tot R i satisfà una de les dues condicions següents:

1. La corba c és difeomorfa a un cercle. És a dir, existeix un $T > 0$ tal que $c(s_1) = c(s_2)$, si i només si, $s_1 - s_2$ és un múltiple de T .
2. La corba c és difeomorfa a una recta real. És a dir, c és injectiva i no té punts d'acumulació quan s tendeix a $\pm\infty$.

Així doncs, si tenim un punt zero inicial que sigui un valor regular podem assegurar que existeix una única corba $c : I \rightarrow R$ definida en un interval màxim d'existència I que satisfà el problema de valor inicial definit al punt (2.4).

3. Conceptes bàsics del mètode Predictor-corrector

La idea en els mètodes PC és traçar la corba numèricament generant punts x_i , $i = 1, 2, \dots$ satisfent que la norma de H (4) satisfaci: $\|H(x_i)\| < \epsilon$, per a algun $\epsilon > 0$.

Com el seu nom indica, els passos que realitza són el *predictor step* i el *corrector step*.

Suposem que comencem en un punt $u_0 = (x_0, \lambda_0)$ que pertany a la corba $c(s)$ per a algun s . En el *predictor step* el que fem és calcular la tangent a la corba $c(s)$ en el punt s i ens movem un increment Δs en la direcció d'aquesta tangent. Així doncs sempre ens mourem en el sentit de la corba; ara bé, aquest nou punt al que arribem (y_s, λ_s) on $\lambda_s = \lambda_0 + \Delta s$, no pertany a la corba a menys que es tracti d'una recta, aquí és on entra en joc el següent pas. En el *corrector step* el que fem es caure sobre la corba $c(s)$, és a dir, tracem la perpendicular de la recta tangent desde el nou punt $(y_s, \lambda_s) = (x_0 + \Delta s_x, \lambda_0 + \Delta s_\lambda)$, i agafem aquell punt que satisfà l'equació (4); aquest doncs, serà el nou (x_1, λ_1) .

Gràficament, es pot veure més clarament a partir de la figura 3.

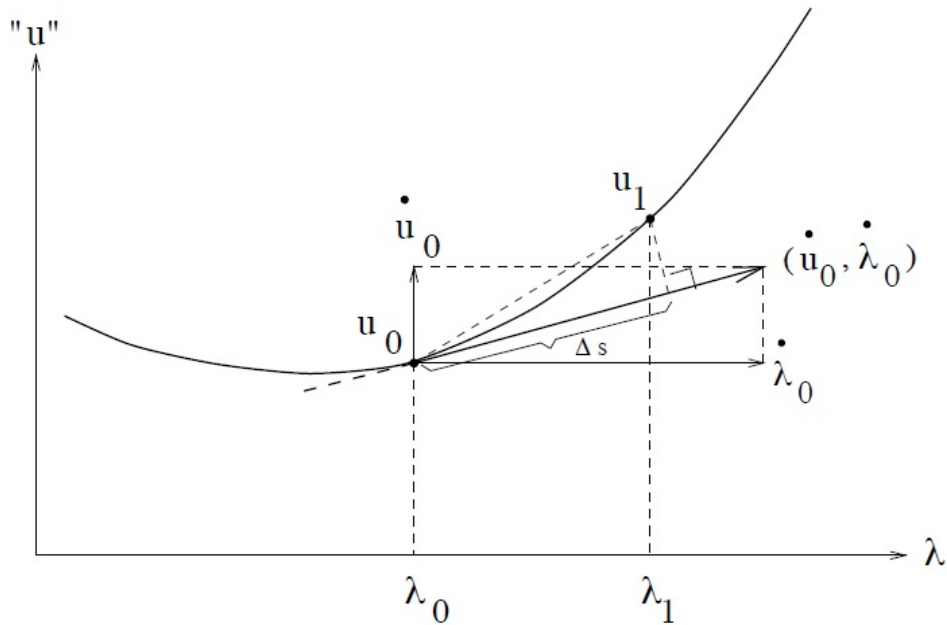


Figura 3: Foto extreta de [1]. Com es pot veure en el gràfic, el nou punt $u_1 = (x_1, \lambda_1)$ esdevé de traçar la perpendicular de la tangent desde el punt arribat en el pas predictor.

4. Implementació del mètode Predictor-corrector

Un cop vist el funcionament teòric dels mètodes PC, estudiarem amb més detall com funcionen i com s'arriba a cada pas del procés.

Sigui $H : R^{N+1} \rightarrow R^N$ una funció continua. Considerem $x \in R^{N+1}$ on $x = (x_0, x_1, x_2, \dots, x_N) = (\lambda, x_1, x_2, \dots, x_N)$ i λ respresenta el paràmetre de continuació. Així doncs, H vindrà donada de la forma:

$$H = \begin{pmatrix} f_1(\lambda, x_1, x_2, \dots, x_N) \\ f_2(\lambda, x_1, x_2, \dots, x_N) \\ \dots \\ \dots \\ f_i(\lambda, x_1, x_2, \dots, x_N) \\ \dots \\ \dots \\ f_{N-1}(\lambda, x_1, x_2, \dots, x_N) \\ f_N(\lambda, x_1, x_2, \dots, x_N) \end{pmatrix}$$

on $(f_i)_{i=1, \dots, N} : R^{N+1} \rightarrow R$ representen N funcions contínues, ja que H és continua. Considerem el paràmetre arc donat per:

$$(\Delta s)^2 = \sum_{i=0}^N (\Delta x_i)^2$$

Considerem la derivada parcial respecte el paràmetre arc de cada coordenada de x , x_i com:

$$\frac{dx_i}{ds} = \frac{\pm A_i}{\sqrt{\sum_{i=0}^N A_i^2}} \quad (9)$$

El signe positiu o negatiu que acompanya al numerador, es decidirà depenent del sentit en el que volgum recórrer la corba. Per altra banda, el terme A_i es correspon al determinant de la Jacobiana de H sense les derivades de la variable i -èssima, és a dir, sense la columna i -èssima (10).

$$A_i = \begin{vmatrix} \frac{df_1}{dx_0} & \frac{df_1}{dx_1} & \cdots & \frac{df_1}{dx_{i-1}} & \frac{df_1}{dx_{i+1}} & \cdots & \frac{df_1}{dx_N} \\ \frac{df_2}{dx_0} & \frac{df_2}{dx_1} & \cdots & \frac{df_2}{dx_{i-1}} & \frac{df_2}{dx_{i+1}} & \cdots & \frac{df_2}{dx_N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{df_{i-1}}{dx_0} & \frac{df_{i-1}}{dx_1} & \cdots & \frac{df_{i-1}}{dx_{i-1}} & \frac{df_{i-1}}{dx_{i+1}} & \cdots & \frac{df_{i-1}}{dx_N} \\ \frac{df_i}{dx_0} & \frac{df_i}{dx_1} & \cdots & \frac{df_i}{dx_{i-1}} & \frac{df_i}{dx_{i+1}} & \cdots & \frac{df_i}{dx_N} \\ \frac{df_{i+1}}{dx_0} & \frac{df_{i+1}}{dx_1} & \cdots & \frac{df_{i+1}}{dx_{i-1}} & \frac{df_{i+1}}{dx_{i+1}} & \cdots & \frac{df_{i+1}}{dx_N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{df_{N-1}}{dx_0} & \frac{df_{N-1}}{dx_1} & \cdots & \frac{df_{N-1}}{dx_{i-1}} & \frac{df_{N-1}}{dx_{i+1}} & \cdots & \frac{df_{N-1}}{dx_N} \\ \frac{df_N}{dx_0} & \frac{df_N}{dx_1} & \cdots & \frac{df_N}{dx_{i-1}} & \frac{df_N}{dx_{i+1}} & \cdots & \frac{df_N}{dx_N} \end{vmatrix} \quad (10)$$

Notar que la matriu anterior és una matriu formada per N files i N columnes, ja que tenim $N+1$ variables a causa d'introduir-hi el paràmetre de continuació λ . Un cop definits els termes anteriors, anem a concretar els passos dels mètodes PC.

4.1 Predictor-step

Suposem doncs, que inicialitzem el procés en un punt $u_0 = (\lambda_0, x) = (\lambda_0, x_1, x_2, \dots, x_N)$, $x \in R^N$, que considerem solució de H i per tant, $\|H(u_i)\| < \epsilon$, $H : R^{N+1} \rightarrow R^N$. Si u_i és un punt regular, aleshores els resultats de la secció (2.3) poden ser aplicats i per tant, existeix una única corba $c : I \rightarrow R$ definida en el interval màxim d'existència de I , que satisfà el problema de valor inicial:

$$\begin{cases} u' = t(H'(u)) \\ u(0) = u_0 \end{cases}$$

Primer de tot, en el *pas predictor* el que farem és moure'ns a través de la tangent $t(H'(u))$ amb un pas Δs :

$$v_{i+1} = u_i + t(H'(u_i))\Delta s$$

Per tant, arribarem a un nou punt v_{i+1} amb coordenades $v_{i+1} = (x_{0p}, x_{1p}, x_{2p}, \dots, x_{Np}) = (\lambda_{0p}, x_{1p}, x_{2p}, \dots, x_{Np})$ on cada coordenada és calculada a partir de la equació (9) de la següent forma:

Predictor-step

$$\begin{aligned}
x_{0p} &= x_0 + \frac{dx_0}{ds} * (-1)^0 * \Delta s \\
x_{1p} &= x_1 + \frac{dx_1}{ds} * (-1)^1 * \Delta s \\
&\dots \\
x_{ip} &= x_i + \frac{dx_i}{ds} * (-1)^i * \Delta s \dots \\
x_{Np} &= x_N + \frac{dx_N}{ds} * (-1)^N * \Delta s
\end{aligned} \tag{11}$$

Cal notar que com més petit sigui el pas, més ràpidament convergirà, però de la mateixa manera, més lentament avançarem resseguint la corba.

4.2 Corrector-step

Un cop hem avançat a partir de la tangent, ens trobarem en un punt v_{i+1} , $i \geq 0$, que no serà solució de les equacions a no ser que la corba sigui una recta, és a dir, no es complirà que $\|H(v_{i+1})\| < \epsilon$, i per tant, $v_{i+1} \notin c(s)$. Per no fer un abús de notació, a partir d'ara, direm $y_k := v_{i+1}$.

En el *pas corrector* el que volem és corregir el pas anterior, és a dir, a partir del punt on hem arribat rectificar-lo fins a caure sobre la corba de solucions $c(s)$. Per a aconseguir-ho, el que farem és traçar la perpendicular de la tangent des del punt y_k i recórrer aquesta perpendicular fins a tallar amb la corba $c(s)$. Així doncs, trobant-nos en el punt: $y_k = (x_{0p}, x_{1p}, x_{2p}, \dots, x_{Np}) = (\lambda_{0p}, x_{1p}, x_{2p}, \dots, x_{Np})$, fins que no es compleixi $\|H(y_k)\| < \epsilon$, aplicarem reiteradament el procés següent:

$$y_{k+1} = y_k - Df(y_k)^T [Df(y_k) Df(y_k)^T]^{-1} f(y_k) \tag{12}$$

Notar que el pas corrector, és anàleg al clàssic mètode de Newton però amb la diferència que es substitueix la inversa clàssica per el terme: $Df(y^k)^T [Df(y^k) * Df(y^k)^T]^{-1}$, on $Df(y^k)$ representa la jacobiana de H evaluada en el punt y^k [5].

Repetint aquest procés iterativament arribarem a la solució desitjada pertanyent a la seqüència de solucions $c(s)$.

Per a comprovar que el mètode funciona, hem agafat una funció senzilla de la qual ja en coneixem les solucions i hi hem aplicat el mètode d'incrustació i el mètode predictor-corrector. D'aquesta manera, podrem veure més clarament les diferències dels resultats.

Hem agafat com a equació:

$$H(x, \lambda) = \lambda x(x-1)(x-3) + (1-\lambda)(x-2) \tag{13}$$

On agafem $F(x) = x(x-1)(x-3)$, $G(x) = x-2$ seguint la notació de (4). És evident que sabem trobar els zeros tant de F com de G , però és un exemple clar en el que podem veure la diferència entre els mètodes presentats anteriorment.

Primer de tot, hem aplicat el mètode d'incrustació, inicialitzant en un punt $H(x_1, 1)$ amb l'objectiu d'arribar a obtenir la solució $H(\bar{x}, 0)$ que fós solució de $G(x)$. Ara bé, al només variar el paràmetre λ en un sentit, és a dir, moure de manera decreixent el paràmetre λ de 1 a 0, perdem la continuació de la corba (Figura 4).

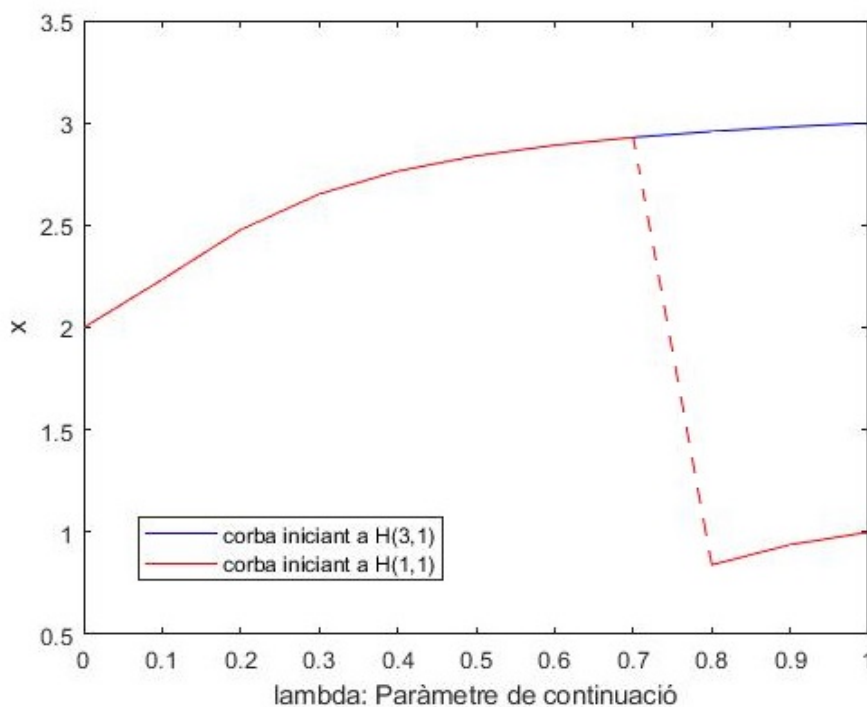


Figura 4: Podem veure que aproximadament aprop de $\lambda = 0.8$ la corba fa un salt i es desvia cap a l'altra corba, per tant deixem de tenir la continuïtat, alhora que perdem la seqüència de solucions properes.

Ara bé, quan apliquem el mètode PC resseguint la corba a partir del paràmetre arc ens mantenim en una seqüència de solucions que són properes tot i canviar el sentit de λ , tal com volíem obtenir (Figura 5). Per a no marxar-nos dels valors de λ entre 0 i 1, hem afegit en la programació la condició que el mètode pogués el paràmetre de continuació λ fins a topa-se amb una de les dues parets laterals, formades per $\lambda = 0$ i $\lambda = 1$ respectivament.

Així doncs, podem observar que a partir dels mètodes PC podem resseguir les solucions en les que hi ha un retrocés enrere, evitant d'aquesta manera que una solució es perdi; i per tant, hem corregit els errors que ens trobavem quan aplicàvem el mètode d'incrustació.

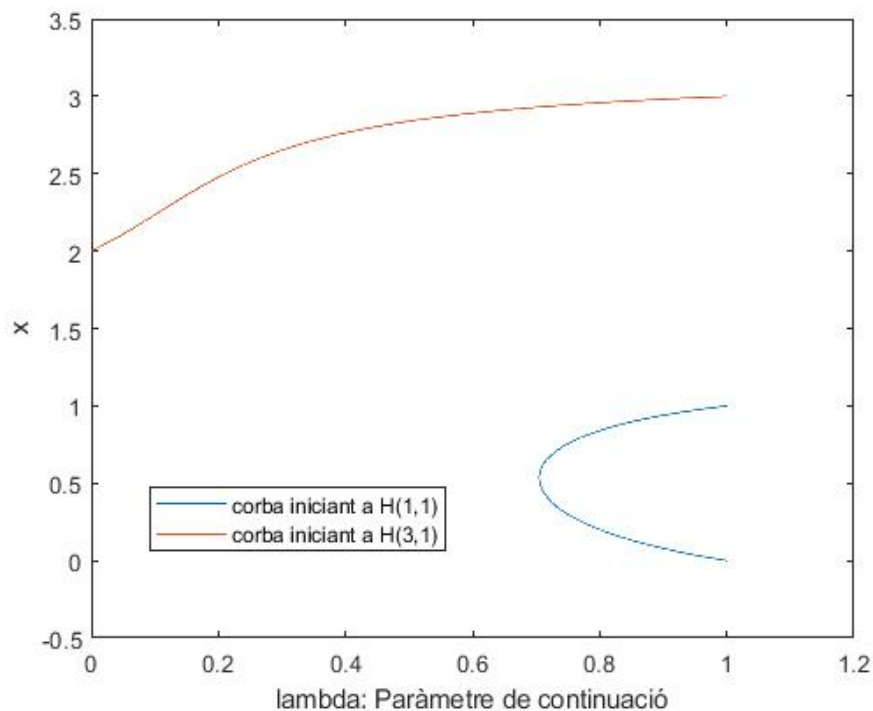


Figura 5: Seqüència de solucions de l'equació (13) traçades a partir del mètode predictor-corrector.

5. Aplicació a un problema biofísic

Un cop vist el funcionament del mètode, l'objectiu és poder-ho aplicar a un cas més concret i d'aquí treure'n conclusions.

5.1 Introducció del problema

El calci és un missatger intracel·lular responsable de totes les funcions cel·lulars com per exemple la contracció de les cèl·lules cardíaques o l'ajustament del metabolisme, i a vegades presenta dinàmiques espai-temporals com per exemple les oscil·lacions de calci. Per a estudiar l'esdeveniment d'aquestes dinàmiques més complexes, és important entendre primerament la homeòstasi del calci, és a dir, els valors d'equilibri que pren en funció del sistema.

En els models més senzills, es considera que la cèl·lula està dividida en diferents compartiments (figura 6) que es corresponen a diferents concentracions de calci. En el nostre cas, hem dividit la concentració de calci en 3 compartiments: c_i , c_j , c_d . La concentració c_d es correspon a la concentració aprop de la membrana de la cèl·lula, la concentració c_j es correspon a la concentració de calci aprop del reticle sarcoplasmàtic (SR) i c_i representa la concentració de calci a l'interior del citosol (dintre de la cèl·lula). Per altra banda, podem observar 3 corrents, que es corresponen amb el transport de calci entre els diferents compartiments.

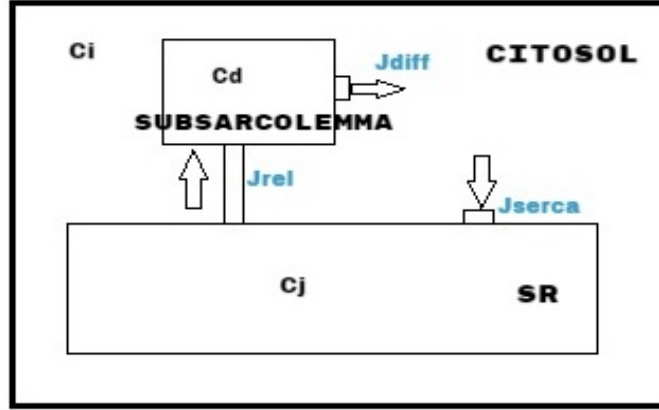


Figura 6: Cèl·lula sense intercanvi de calci amb l'exterior. Està formada per 3 compartiments i els 3 canals d'intercanvi de calci respectius.

Donades aquestes condicions i suposant que no hi ha intercanvi de calci amb l'exterior, definim la quantitat de calci total com Q_T . Així doncs, si no hi intervinguessin altres factors, podem deduir que Q_T ve definida per la fórmula següent:

$$Q_T = v_i c_i + v_d c_d + v_{jsr} c_j \quad (14)$$

on els coeficients v_i , v_d , v_{jsr} representen els coeficients de volum respectius per a cada compartiment. Ara bé, no tot el calci pot ser transportat d'un compartiment a un altre ja que certa quantitat de calci s'adhereix a proteïnes que conté la cèl·lula, el que comporta que no tota la concentració de calci sigui lliure. Si afegim les quantitats de calci que no són lliures, l'equació anterior queda de la següent manera:

$$Q_T = v_i c_i + v_i \frac{B c_i}{K_B + c_i} + v_d c_d + v_{jsr} c_j + v_{jsr} \frac{B_{SQ} c_j}{K_{SQ} + c_j} \quad (15)$$

on B i B_{SQ} són les quantitats totals de proteïnes que s'uneixen al calci (calcium buffers) al citosol i SR, respectivament. Un cop introduïts en el model de cèl·lula que tenim, definim els canals de corrent com: J_{SERCA} := corrent d'intercanvi de calci entre el citosol i el SR, J_{diff} := corrent d'intercanvi de calci entre la membrana i el citosol, i per últim, J_{rel} := representa la corrent d'intercanvi de calci entre el SR i la membrana de la cèl·lula. Donades les condicions anteriors, les equacions per les quals es regeix la variació de la concentració de calci d'aquesta cèl·lula són:

$$\frac{dc_d}{dt} = J_{rel} - J_{diff} \quad (16)$$

$$\frac{dc_j}{dt} = -\frac{v_d}{v_{jsr}} J_{rel} + J_{SERCA} \quad (17)$$

El valor de c_j s'obté resolent l'equació algebraica (15). Partint d'aquí, el nostre objectiu serà trobar les possibles concentracions c_i , c_d , c_j a partir de la concentració Q_T .

5.2 Paràmetres del model del problema

Considerant el model determinístic sobre el calci en una cèl·lula cardíaca que hem vist anteriorment, agafem els paràmetres per estudiar-ne l'existència i estabilitat de diferents solucions. Els paràmetres del model venen donats per la taula de la figura 7.

Parameters		
Parameter	Units	Value
v_d	μm^3	0.001
v_i	μm^3	0.45
v_{jsr}	μm^3	0.02
τ_d	ms	0.008
g_{up}	$\mu M / ms$	0.5
K_S	μM	0.2
g_r	ms^{-1}	50
K_O	μM	15
k_i	$ms^{-1} \mu M^{-1}$	0.01
k_r	ms^{-1}	0.01
B	μM	80
K_B	μM	0.5

Figura 7: Presenta els valors que farem servir en el model de la nostra cèl·lula.

Dels paràmetres de la taula: g_r representa la força del corrent d'alliberament J_{rel} , g_{up} és la força de bombeig J_{SERCA} , v_i , v_d , v_{jsr} són els volums associats a cada compartiment, K_s és la concentració ($10^{-6} mol/L$) a la que el corrent SR es tanca, K_o és la concentració a la que SR obra i τ representa la constant de temps de difusió.

Donats els paràmetres anteriors, les corrents J_{SERCA} , J_{diff} , J_{rel} venen donades per:

$$J_{diff} = \frac{c_d - c_i}{\tau} \quad (18)$$

$$J_{rel} = g_r \frac{cd^2}{K_o^2 + c_d^2} (c_j - c_d) \quad (19)$$

$$J_{SERCA} = \frac{v_i}{v_{jsr}} g_{up} \frac{c_i^2}{K_s + c_i^2} \quad (20)$$

Agafant aquestes corrents i substituint-les en les equacions (16) i (17), juntament amb l'equació (15), ens queda el següent sistema de 3 equacions per als punts d'equilibri:

$$0 = g_r \frac{c_d^2}{K_o^2 + c_d^2} (c_j - c_d) - \frac{c_d - c_i}{\tau} \quad (21)$$

$$0 = \frac{-v_d}{v_{jsr}} g_r \frac{c_d^2}{K_o^2 + c_d^2} (c_j - c_d) + \frac{v_i}{v_{jsr}} g_{up} \frac{c_i^2}{K_s + c_i^2} \quad (22)$$

$$c_i + \frac{Bc_i}{K_B + c_i} = \frac{1}{v_i} (Q_T - v_d c_d - v_{jsr} c_j - v_{jsr} \frac{B_{SQ} c_j}{K_{SQ} + c_j}) \quad (23)$$

Com podem comprovar, ens queda un sistema de 3 equacions amb 4 incògnites: c_i , c_d , c_j , Q_T ; ara bé, com estem suposant que no hi ha intercanvi de calci amb l'exterior, la quantitat de calci total Q_T esdevé una constant, i per tant, a partir de les 3 equacions algebraïques anteriors i aplicant els mètodes de continuació apresos agafant com a paràmetre de continuació Q_T , podrem trobar les concentracions de calci de la cèl·lula en funció de la concentració total.

5.3 Concentracions de calci del model del problema

Per simplicitat considerarem primer el cas en que no hi ha *buffers*, és a dir, el terme B_{SQ} és 0. Com aquest terme està al numerador, l'últim terme de l'equació (23) queda cancel·lat. Aleshores utilitzant el clàssic mètode de substitució, es fàcil obtindre de les equacions (23), (22):

$$c_d = c_i + \frac{v_i}{v_d} \frac{\tau g_{up} c_i^2}{K_s^2 + c_i^2} \quad (24)$$

$$c_j = \frac{1}{v_{jsr}} [Q_T - v_d c_d - v_i (c_i + \frac{Bc_i}{K_B + c_i})] \quad (25)$$

En aquestes dues últimes equacions tot terme depèn del terme c_i , així doncs, substituint-les en la primera equació (21) podem reduir el problema a una equació amb només una incògnita en funció de Q_T de la forma $f(c_i; Q_T) = 0$. Arribats a aquest punt, aplicarem els conceptes apresos sobre els mètodes de continuació PC agafant com a paràmetre de continuació Q_T , d'aquesta manera obtindrem per a cada valor de Q_T un valor c_i que resol l'equació.

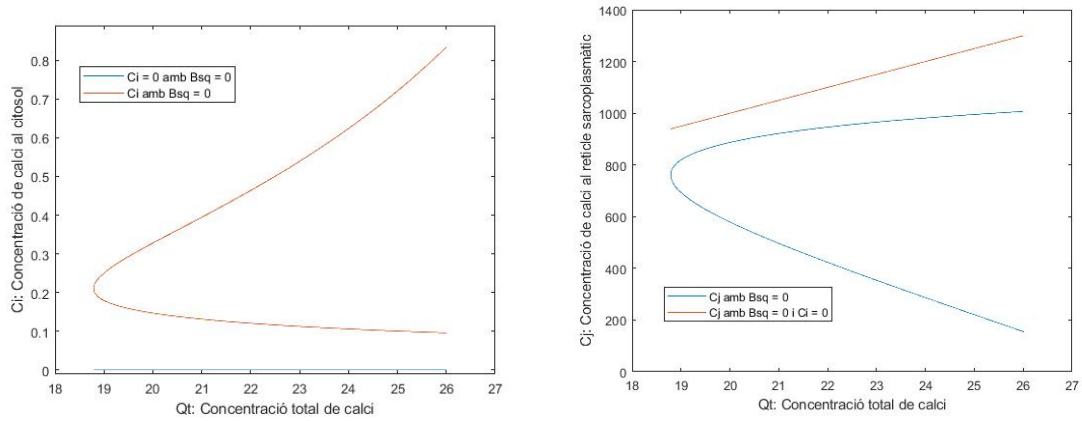


Figura 8: Seqüència de solucions de les concentracions c_i , c_j en funció del paràmetre de continuació Q_T en una cèl·lula amb $B_{SQ} = 0$, és a dir, sense buffers.

Podem veure que hi ha punts on tenim més d'una possible solució (figura 8), i per tant, podem tindre varies concentracions possibles. Per exemple, agafant $Q_T = 22$, tenim tres concentracions possibles per a $c_i = 0.4639$, $c_i = 0.1211$ i $c_i = 0$, per tant, tres possibles solucions de c_d , c_j també. Així doncs, en aquest cas tindriem: $x_1 = (c_i, c_j, c_d) = (0.4639, 423.1693, 1.9818)$, $x_2 = (0.1211, 946.2430, 0.6041)$ i $x_3 = (0, 1100, 0)$, respectivament.

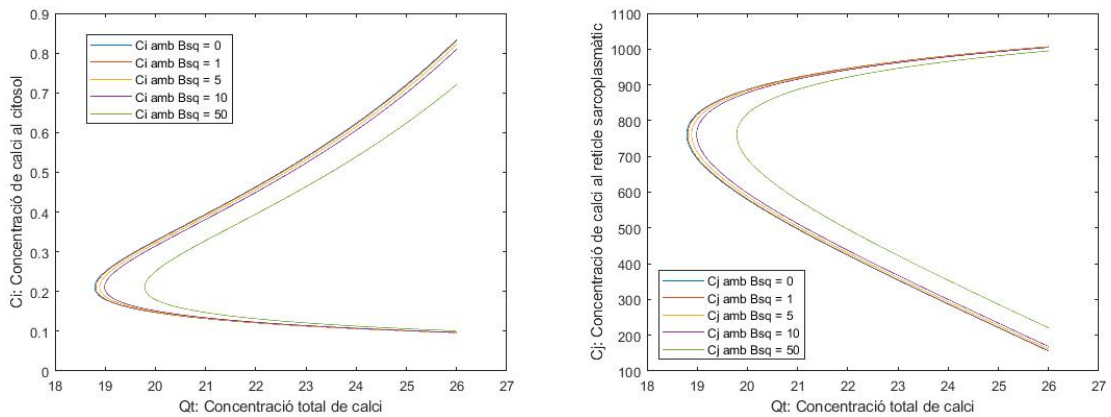


Figura 9: Seqüència de solucions de la concentració c_i en funció del paràmetre de continuació Q_T en una cèl·lula variant el paràmetre B_{SQ} .

Considerem ara el cas en que hi ha un nombre positiu de proteïnes a les quals s'adhereix una certa quantitat de calci.

Podem veure que si la quantitat total de calci en la cèl·lula és baixa, aleshores existeix una única solució en la que gairebé tot el calci es troba en $SR(c_i$ baix i c_j alt). A mesura que augmentem el calci total Q_T apareixen règims amb més solucions, donant biestabilitat entre un estat amb nivell baix de calci en el citosol i un altre amb un nivell alt. Això és degut a que, al augmentar c_i ,

augmenta c_d i això fa que s'obri la corrent J_{rel} , que ahora és l'encarregada de mantindre un nivell alt de calci en el citosol. Si canviem la concentració de buffers (figures 9 i 10) l'efecte és similar, existeix menys calci lliure (ja sigui en el citosol o en el SR) i per tant es necessita una quantitat més gran de Q_T per a que aparegui la branca superior.

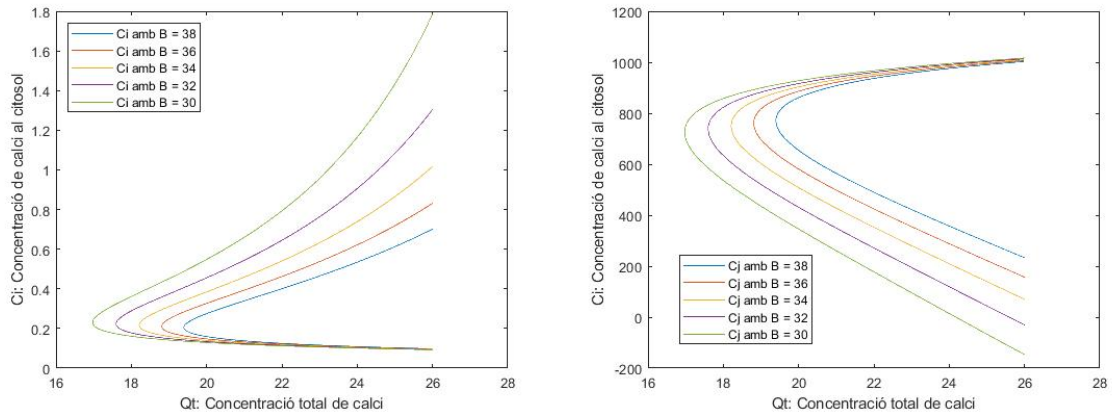


Figura 10: Seqüència de solucions de les concentracions c_i, c_j en funció del paràmetre de continuació Q_T en una cèl·lula, variant el paràmetre B .

6. Conclusions

En aquest projecte he estudiat els conceptes bàsics sobre els mètodes de continuació i he descrit els passos a realitzar per a aconseguir automatitzar els mètodes numèricament. Partint d'un mètode completament senzill però alhora efectiu com era el mètode d'incrustació, hem anat profunditzant i analitzant els seus errors per a poder corregir-los. Per a aconseguir-ho, ens hem endinsat en el món dels paràmetres de continuació i de les corbes, estudiant les condicions que havien de satisfer aquestes corbes amb la finalitat de poder traçar-les i poder-les fer servir com a mètode de continuació de solucions. Tot seguit hem estudiat els mètodes PC coneguts com *Predictor-corrector methods*, i hem analitzat els passos que aquest realitza intentant replicar-los amb MATLAB. Per últim, un cop assolida l'automatització dels mètodes PC mitjançant MATLAB, hem aplicat aquest procés sobre un cas biofísic agafant com a paràmetre de continuació la concentració total de calci Q_T d'una cèl·lula, hem buscat les possibles combinacions de concentracions de calci de cada compartiment de la cèl·lula traçant la corba de solucions a partir dels mètodes PC i d'aquesta manera n'hem pogut extreure conclusions.

Per a possibles ampliacions del treball podriem analitzar bifurcacions i solucions periòdiques, el que permetria analitzar l'aparició i estabilitat de solucions oscil·ladores en el model biofísic. De la mateixa manera, hem buscat solucions del sistema estacionari, per tant també podriem investigar com es comporta i quines variacions adquireixen aquestes concentracions quan poden variar en funció del temps. Finalment, es podria aprofundir tractant l'altre mètode de continuació anomenat: Piece-wise linear methods.

Referències

- [1] Eusebius Doedel. *An introduction to Numerical Continuation Methods with Application to some Problems from Physics*, Cuzco, Peru, 2015.
- [2] Eugene L. Allgower Kurt Georg. *Introduction to Numerical Continuation Methods*.
- [3] John Burkardt. *The Continuation Method for Algebraic Nonlinear Equations*, Department of Scientific Computing, Florida State University.
- [4] F. A. Ficken. *The continuation method for functional equations.*, New York University and the University of Tennessee, 1951.
- [5] Carles Simó. *On the Analytical and Numerical Approximation of Invariant Manifolds.*, Departament de Matemàtica Aplicada i Anàlisi, Universitat de Barcelona.

A. Codis de MATLAB

Finalment, afegirem els codis principals que han estat creats per a realitzar els processos.

El següent codi, és l'encarregat de dibuixar les seqüències de punts solució a partir dels mètodes PC; va lligat a tot un conjunt de subcodis que també afegirem.

```
x0 = input('Aproximacio_inicial_x0: ') %1
qt = input('Aproximacio_inicial_Qt: ') %26
B = input('Valor_de_B: ') %36
Bsqr = input('Valor_de_Bsq: ') %1
limit = input('Valor_on_volem_aturar_el_proces: ') %26
[vr, xf, niter] = corrector(x0, qt, B, Bsqr, @avaluafuncio, @derivadanumerica);

[xk, lk] = predictor(x0, qt, B, Bsqr, @avaluafuncio);

i = 0;
matriu_x = zeros(2,300);
while(lk < limit)
[vr, xf, niter] = corrector(xk, lk, B, Bsqr, @avaluafuncio, @derivadanumerica);
x0 = xf;
l = lk;
i = i + 1;
[xk, lk] = predictor(x0, l, B, Bsqr, @avaluafuncio);
matriu_x(1,i) = xk;
matriu_x(2,i) = lk;
end
plot(matriu_x(2,:), matriu_x(1,:))

xlabel('Qt: Concentracio_total_de_calci')
ylabel('Ci: Concentracio_de_calci_al_citosol')
legend('Ci_amb_B=36', 'Location', 'northwest')

figure(2);

matriu_y = zeros(2,300);
for j = 1:i
ci = matriu_x(1,j);
cd = ci + 450*((0.004*ci^2)/(0.04+ci^2));
matriu_y(1,j) = 50*(qt - 0.001*cd - 0.45*ci - (36*ci/(0.5+ci)));
matriu_y(2,j) = matriu_x(2,j);
end
plot(matriu_y(2,:), matriu_y(1,:))

xlabel('Qt: Concentracio_total_de_calci')
ylabel('Cj: Concentracio_de_calci_al_reticle_sarcoplasmatic')
legend('Cj_amb_Bsq=0', 'Location', 'southwest')
```

El següent codi, és l'encarregat de dibuixar les seqüències de solucions a partir del mètode d'incrustació.

```
%% Proceso de l = 0 a l = 10

x0 = input('Aproximacio_inicial_x0: '); %0
sol = zeros(1,10);
```

```

for l = 0:0.1:1
[vr,xf,niter] = newton_tol(x0,l,@avaluafuncio_senzilla,@derivadanumerica);
sol(int8(l*10+1)) = xf;
x0 = xf;
end

x = 0:0.1:1
plot(x, sol, 'b')
hold on
% Proceso de l = 1 a l = 0
x0 = input('Aproximacio_inicial_x0: '); %1.1
sol = zeros(1,10);
for l = 1:-0.1:0
i = 1 - l;
[vr,xf,niter] = newton_tol(x0,l,@avaluafuncio,@derivadanumerica);
sol(int8(l*10+1)) = xf;
x0 = xf;
end

plot(x, sol, 'r')
xlabel('lambda: Parametre_de_continuacio')
ylabel('x')
legend('corba_iniciant_uuH(3,1)', 'corba_iniciant_uuH(1,1)', 'Location', 'northwest')
hold off

```

Funció sobre la qual es volen aplicar els mètodes:

```

function [res] = avaluafuncio(ci,qt,B,Bsq)
Ksq = 0.63;
cd = ci + 450*((0.004*ci^2)/(0.04+ci^2));
caux = 50*(qt - 0.001*cd - 0.45*ci - (B*ci/(0.5+ci)));
b = Ksq + Bsqr - caux;
c = caux*(-Ksq);
cj = (-b + sqrt(b^2 - 4*c))/2;%cambiar signo dependiendo si cogemos la positiva
o negativa
res = 50*(cd^2/((15^2)+(cd^2)))*(cj-cd)- ((cd-ci)/0.008);
end

```

Pas predictor dels mètodes PC:

```

function [xk,lk] = predictor(x0,l,B,Bsq,f)

h = 1e-5;
s = 0.0001;

dfx0 = [[f(x0+h,l,B,Bsq)] - [f(x0,l,B,Bsq)]]/h;
dlx0 = [[f(x0,l+h,B,Bsq)] - [f(x0,l,B,Bsq)]]/h;

xk = x0 + (dlx0 / sqrt((dfx0)^2 + (dlx0)^2))*(s);
lk = l + (dfx0 / sqrt((dfx0)^2 + (dlx0)^2))*(-s);

end

```

Pas corrector dels mètodes PC:

```

function [vector_r,xf,niter] = corrector(x0,l,B,Bsq,f,df)

h = 0.0005;
niter = 1;
found = 0;
tol = 0.5e-2;
while(found == 0)
    A = df(x0,l,B,Bsq,h,f);
    x1 = x0-(A'*inv(A*(A')))*f(x0,l,B,Bsq);
    rk = abs((x0-x1)/x1);
    vector_r(niter) = rk;
    if (rk < tol)
        if (abs(f(x1,l,B,Bsq))< tol)
            found = 1;
        end
    end
    niter = niter+1;
    x0 = x1;
end

xf = x0;
yf = f(xf,l,B,Bsq);
niter = niter - 1;
end

```

Mètode de Newton:

```

function [vector_r,xf,niter] = newton_tol (x0,l,f,df)

%indiquem la h
h = 0.0005;
niter = 1;
found = 0;
tol = 0.5e-10;% el 10 indica el nombre de xifres significatives
while(found == 0)

    x1 = x0-(f(x0,l)/df(x0,l,h,f));
    rk = abs((x0-x1)/x1);
    vector_r(niter) = rk;
    if (rk < tol)
        if (abs(f(x1,l))< tol)
            found = 1;
        end
    end
    niter = niter+1;
    x0 = x1;
end

xf = x0;
yf = f(xf,l);
niter = niter - 1;
end

```

Derivada numèrica:

```
function [dfx0] = derivadanumerica(ci,qt,B,Bsq,h,f)
dfx0 = [[f(ci+h,qt,B,Bsq)] - [f(ci,qt,B,Bsq)]]/h;
end
```